Comparação entre abordagens escaláveis para o processamento de conjuntos de dados textuais

Comparison of scalable approaches to process textual data sets

Gustavo de Paula Avelar [1]
Murilo Coelho Naldi [2]

Data de submissão: 28/06/2016, Data de aceite: 12/02/2017

Resumo: *Data Analytics* é um conceito voltado a análise de grandes quantidades de dados em busca de padrões e informações relevantes. A manipulação desses dados é complexa e exige métodos automáticos capazes de processar grandes volumes de dados exigindo poder computacional para obtenção de informações em tempo hábil. O modelo de programação *MapReduce* surgiu para auxiliar a distribuição desses problemas entre várias máquinas, melhorando a eficiência em seu processamento. As plataformas *Apache Hadoop* e *Spark* possibilitam a utilização deste paradigma em ambientes de *hardware commodities*. O agrupamento de dados tem como objetivo determinar um conjunto finito de categorias para descrever um conjunto de dados de acordo com as características similares dos objetos do conjunto de dados. Diferentes estratégias para pré-processamento influenciam os resultados da etapa de agrupamento de dados. Deste modo, este trabalho trata do estudo de diferentes métodos de pré-processamento de documentos textuais, visando alcançar representações que proporcionem bons resultados à etapa de agrupamento. Nele, propomos uma abordagem para seleção de atributos embasado no algoritmo *Latent Dirichlet Allocation* (LDA).

[1]Instituto de Ciências Exatas e Tecnológicas, Universidade Federal de Viçosa — Campus de Rio Paranaíba, Rodovia MG 230 - Km 7, Caixa Postal 22, CEP: 38810-000, Rio Paranaíba - MG
{gustavo.paula@ufv.br}
[2]Departamento de Computação, Universidade Federal de São Carlos, Rod. Washington Luís - Km 235 Caixa Postal 676, CEP 13565-905, São Carlos-SP
{naldi@dc.ufscar.br}

Abstract: Data Analytics is an emerging concept intended to describe the searching of patterns and relevant information in large amounts of data. Dealing with these data is complex and requires automatic methods for processing such amount of data and demand high computation power to extract information in a timely manner. The programming model MapReduce was proposed to assist the distribution of these problems between multiple machines, improving efficiency in processing. Platforms Apache Hadoop and Spark allow the use of this paradigm in an ambient with commodities hardware. Clustering aims to determine a finite set of categories to describe a set of data according to the similar features of objects from a dataset. Different strategies for pre-processing influence the results of the clustering step. This paper deals with the study of different methods of pre-processing of textual documents in order to achieve representations that provide good results to the clustering step. We propose an approach for the selection of attributes grounded in the algorithm Latent Dirichlet Allocation (LDA). The algorithms were evaluated on different datasets of real data, the experimental results indicate that the accuracy achieved by the algorithms on real data varies according to the number of objects, determined by the curse of dimensionality, and the feature selection approach proposed by the authors improves the clustering step.

***P*alavras-chave:** Análise de Dados, Agrupamento, pré-processamento, seleção de atributos.

***K*eywords:** Data Analytics, clustering, pre-processing, feature selection.

1 Introdução

Data Analytics (1) é um conceito emergente voltado para investigação de grandes quantidades de dados utilizando algoritmos do estado da arte com o objetivo de buscar padrões e obter informações relevantes sobre os conjuntos de dados. A aquisição de informações sobre os dados é exigida de forma eficiente devido ao crescimento rápido e contínuo dos volumes de dados.

A análise de dados utilizando procedimentos manuais é descartada devido aos grandes volumes de dados a serem investigados. A intensidade que estes dados estão sendo gerados afetam também a utilização de sistemas de gerenciamento de banco de dados (SGBD) tradicionais o que impõe a necessidade por implementações de métodos automáticos para processamento e análise em ambientes escaláveis tendo em vista a extração de conhecimento (2).

Uma solução para esse efeito é a implementação de programas utilizando o modelo de programação *MapReduce* (3) que possibilita a aplicação de uma solução paralela e dis-

tribuída para gerir e processar grandes quantidades de dados. Este modelo é utilizado pelas plataformas *Apache Hadoop* (4) e *Spark* (5) que estão ganhando cada vez mais força neste cenário.

Ao realizar o processamento de dados reais/textuais é necessário efetuar uma preparação nos dados, onde é fixada uma estrutura padrão para representação dos dados textuais. Diversas técnicas como normalização, categorização, redução de dimensionalidade, seleção de atributos, dentre outras podem ser aplicadas nesta etapa com intuito de atingir melhores resultados ao final do processo (6).

Diversos algoritmos para *Data Analytics* foram desenvolvidos no modelo *MapReduce*. Dentre eles, o *K-means* (7) que é um algoritmo de agrupamento de dados não supervisionado que encontra um conjunto de grupos k desconhecido no qual os dados são compostos. Os grupos são formados por objetos que abrangem características semelhantes ou próximas. A utilização de algoritmos automáticos sobre dados de natureza desconhecida facilita a obtenção de informações relacionadas a dispersão dos objetos do conjuntos de dados. Para a fase de pré-processamento os algoritmos de redução de dimensionalidade Análise de Componentes Principais e *Latent Dirichlet Allocation* podem ser utilizados.

A Análise de Componentes Principais (8) é uma técnica para redução de dimensionalidade que auxilia na formação de uma representação reduzida dos dados com base na combinação linear dos conjuntos de dados. Este algoritmo é aplicado tendo em vista a representação por subespaços com as maiores variâncias dos conjuntos de dados originais. Os componentes principais são encontrados pela escolha dos autovetores que possuem autovalores altos (9). Assim, as relações entre os objetos são mantidas através das características mais relevantes dos conjuntos de dados. O *Latent Dirichlet Allocation* (10) é um algoritmo poderoso para detecção de padrões em documentos textuais usado para redução de dimensionalidade e também para o agrupamento de dados. Este algoritmo determina um subespaço dos dados formado por tópicos sem que haja perda de informações e detalhes da representação original dos documentos.

A utilização de diferentes técnicas para redução de dimensionalidade contribui para obtenção de representações reduzidas dos conjuntos de dados. Esse procedimento torna mais fácil o processamento de documentos textuais que são altamente esparsos, além de tentar descrever as relações entre os objetos com uma quantidade menor de características.

No cenário de exploração das técnicas de redução de dimensionalidade, temos os (11),(12),(13),(14),(15),(16),(17). A maior parte destes trabalhos utilizam técnicas que não foram avaliadas sobre o modelo de programação *MapReduce*, ficando limitados aos recursos de uma máquina.

Este trabalho propõe um estudo comparativo de diversas abordagens desenvolvidas sobre o modelo de programação *MapReduce* para o processamento de conjuntos de dados

reais em contexto escalável. O framework Hadoop permite a utilização de uma à milhares de máquinas, trazendo escalabilidade as abordagens propostas. Nós avaliamos as técnicas *K-means*, Análise dos componentes principais (ACP) e *Latent Dirichlet Allocation* (LDA) quando aplicadas a conjuntos de dados textuais. Nós também propomos e investigamos a seleção de atributos com base nos tópicos computados pelo algoritmo *Latent Dirichlet Allocation*. As técnicas utilizadas são analisadas de forma a observar quais metodologias de pré-processamento (ACP e LDA) geram melhores resultados à etapa de agrupamento de dados (*K-means*) quando aplicada a conjuntos de dados textuais de diferentes proporções de objetos.

O restante deste artigo é organizado do seguinte modo: na Seção 2 são apresentadas as principais concepções que ajudam na compreensão deste trabalho. Na Seção 3 os algoritmos utilizados são apresentados de forma detalhada ao modelo de programação utilizado. Na Seção 4 são mostrados os conjuntos de dados reais utilizados, índices de validação aplicados e os resultados dos experimentos realizados. Por último, é apresentada a conclusão na Seção 5.

2 Trabalhos Relacionados

Na busca pela descoberta do conhecimento (do inglês, *knowledge discovery in databases* — KDD) (18) em documentos textuais é comum a utilizar o modelo *Bag-of-Words* (19, 20) para representação dos conjuntos de dados em um espaço vetorial (do inglês, *Vector Space Model*). Este modelo descreve cada um dos documentos de acordo com as palavras presentes em seu corpo textual, para cada palavra são atribuídos pesos para realçar a sua importância no documento. O peso ou fator de ponderação, pode ser atribuído pela *TF — Term Frequency* (21) ou *TF-IDF — Term Frequency-Inverse Document Frequency* (22, 23). Para a recuperação de informação (RI) o peso *TF-IDF* é bastante utilizado (20) e, portanto, utilizada no desenvolvimento deste trabalho. Outra aspecto importante no processo KDD é a etapa de pré-processamento. Durante a execuçao desta etapa podemos aplicar diversas técnicas como seleção de atributos, redução de dimensionalidade, dentre outras (24).

Seleção de atributos (25) (26) (27) é uma etapa importante realizada durante o pré-processamento e transformação de qualquer conjuntos de dados. Nela é realizada a seleção de um subconjunto de atributos que representam expressivamente os conjuntos de dados, removendo dados que são irrelevantes e que prejudicam o processo KDD. A remoção de atributos insignificantes ao trabalhar com documentos textuais é efetuada aplicando *Stopwords* (28, 29) diminuindo a representação *Bag-of-Words* de grandes conjuntos de dados, que facilita o processamento dos conjuntos de documentos utilizados neste trabalho.

A redução de dimensionalidade (30) consiste em encontrar d dimensões que contém a representação dos dados de forma reduzida obtidas pela combinação linear ou não. A

dimensionalidade de conjuntos de dados textuais é delimitada pela quantidade de palavras nos conjuntos, portanto quanto maior os conjuntos de dados e seu corpo textual, maiores serão as dimensões para analise e processamento.

Muitas dessas técnicas encontram-se implementadas sobre o modelo de programação *MapReduce*. O paradigma *MapReduce* (3) é o modelo utilizado pelos algoritmos comparados neste trabalho, pois possibilita a aplicação de uma solução paralela e distribuída. O modelo de programação *MapReduce* é estruturado por duas fases: *Map* e *Reduce*. Além de possibilitar o processamento adequado do grande volume de dados e utilização de algoritmos para *Data Analytics*, é possível utilizá-lo com *hardware commodities* presentes no mercado (31).

Em (11) foi proposta uma abordagem para o agrupamento de dados utilizando o algoritmo de agrupamento tradicional *K-means* local após a realização da projeção do conjunto de dados textuais trabalhados em espaço de baixa dimensionalidade obtida pelo algoritmo *Locality Preserving Indexing* que preserva a relação semântica entre os documentos da mesma classes. Diferente do autor, nos avaliamos o algoritmo *K-means* da biblioteca *Mahout* desenvolvido no modelo de programação *MapReduce*, em adicional reduzimos as dimensões do conjunto de dados utilizando o resultado obtido pela combinação linear do algoritmo ACP, as avaliações foram feitas utilizando pequenos e grandes conjuntos de dados.

A proposta de (12) demonstra que a técnica para redução de dimensionalidade não supervisionada Análise dos Componentes Principais (ACP) esta intimamente ligada com o método de agrupamento *K-means* também não supervisionado, ambos tentam minimizar o erro quadrático. Embora os autores tenham avaliado a proposta sobre conjuntos textuais pequenos, os resultados obtidos pelo autores levaram a considerar uma avaliação da técnica em ambiente distribuído sobre novos conjuntos de dados que foram produzidos neste trabalho, que são maiores na quantidade de objetos e dimensões.

O trabalho de (13) descreve uma abordagem elaborada com a combinação do algoritmo de redução de dimensionalidade *Linear Discriminant Analysis* e o algoritmo *K-means* encorporados a um *framework* para a seleção do subespaço que possui a maior representação/discriminação dos dados analisados. A redução das dimensões dos conjuntos de dados por este método é feito através da combinação linear de características que separam um ou mais grupos. Diferente do algoritmo *Latent Dirichlet Allocation* que possui um modelo probabilístico para determinar a similaridade entre os dados. O método *Latent Dirichlet Allocation* (LDA) desenvolvido no modelo de programação *MapReduce* é explorado neste trabalho para realizar o agrupamento e seleção de atributos, ao invés do algoritmo *Linear Discriminant Analysis*.

Em (14) foi proposto uma abordagem para redução de dimensionalidade por meio do método *Information gain — InfoGain*. Em seguida, os autores utilizam o algoritmo *K-means* para a etapa de processamento. De forma análoga, a proposta de (15) explora a combinação

entre as técnicas *non-negative matrix factorization (NMF)*, fator ponderação de termos Term Mutual Information (TMI) conectados a uma base de dados lexica, denominada WordNet. Sua proposta explora a extração de termos de acordo com a relação semântica inerente aos documentos processados. Outra características importante, a utilização de processamento paralelo e distribuído sobre o framework *Hadoop*.

O trabalho desenvolvido em (16) utiliza uma abordagem hibrida entre técnicas para seleção de atributos e extração de atributos. Nessa abordagem são utilizadas as técnicas *Term Variance — TV* e *Document Frequency — DF* para ponderação de termos. Em seguida, o algoritmo de Análise de Componentes Principais (ACP) é utilizado para redução de dimensionalidade.

O trabalho de (17) explora a técnica de redução de dimensionalidade Análise de Componentes Principais (ACP) para buscar melhores resultados a etapa de agrupamento de dados com o algoritmo *K-means*. No entanto, utiliza o software MATLAB que requer licença comercial e está limitado aos recursos de uma única máquina.

Embora tenham sido adotadas diversas abordagens para análise de documentos textuais sem a utilização do modelo de programação *MapReduce*, todos estes métodos foram aplicados em conjuntos de dados que possuem uma quantidade pequena de objetos, apesar de possuírem grandes dimensões quando representados pelo modelo *Bag-of-Words*. Nosso trabalho é o primeiro a adotar conjuntos de dados com uma quantidade de documentos maior que os conjuntos de dados textuais utilizados atualmente no campo da pesquisa científica. Outra contribuição deste trabalho é a avaliação de abordagens implementadas no modelo de programação *MapReduce* que facilita análise de grandes conjuntos de dados em contexto escalável. Em adicional é proposto também uma seleção de atributos diretamente relacionado ao resultado obtido pelo algoritmo *Latent Dirichlet Allocation*.

3 Metodologia

Este trabalho foi realizado seguindo o processo KDD(18) com foco no processamento de dados textuais. A execução desse processo envolve cinco etapas essências: coleta de dados; pré-processamento e limpeza; transformação de dados; Mineração de Dados e interpretação e avaliação de resultados.

A coleta de dados pode ser feita em um ambiente experimental controlado com auxilio de um *expert* como também adquirido por meio de diversos sistemas de informação, softwares e paginas web. Após a coleta e elaboração dos conjuntos de dados, passamos para as etapas de pré-processamento e transformação de dados. Neste ponto, são definidos os formatos e a qualidade dos dados durante todo o processo. Essa etapa que consome maior parte do tempo investido no processo (32).

A etapa de Mineração de Dados (MD) (33) busca encontrar informações valiosas sobre grandes quantidades de dados armazenados em SGBD's de forma estruturada, não estruturada. A aquisição de informação é feita de forma preditiva ou descritiva (34). A escolha pelo caminho preditivo permite a criação de modelos através de dados históricos, modelos que são utilizados para prever novos acontecimentos, também conhecido como aprendizado supervisionado. Já o caminho descritivo não sofre influência de dados históricos, descrevendo os padrões inerentes aos dados, conhecido como aprendizado não-supervisionado.

Diversas técnicas foram desenvolvidas para exploração desses dados nesses dois caminhos, tais como: regras de associação (descritivo), agrupamento de dados (descritivo), padrões frequentes (descritivo), métodos de regressão (preditivo), classificação (preditivo) e detecçao de anomalias (preditivo) (34–36).

O foco deste trabalho é a extração de informações relevantes sobre dados textuais utilizando o procedimento de agrupamento de dados, pois a tarefa de agrupamento de dados é um aprendizado não supervisionado. Sendo assim não há necessidade de ter conhecimento prévio do comportamento dos dados. Para extração do informações sobre documentos são necessárias transformações específicas nos corpos textuais dos dados, apresentadas a seguir.

Os conjuntos de dados foram representados no modelo de espaço vetorial (do inglês, *Vector Space Model* (VSM)) (24, 37) que é muito utilizada na área de mineração de texto. Esta estratégia, também conhecida como *Bag-of-Words*, delimita que cada M documentos de texto é formado por T termos (atributos), em que cada $documento_i$ é um vetor de atributos, $documento_i = (a_{11}, a_{12}, a_{13}, a_{14}, \ldots, a_{ij})$. Entretanto, o valor do atributo a_{ij} referente ao termo T_j do $documento_i$ pode ser calculado de diferentes modos. O fator de ponderação *TF-IDF* (abreviação do inglês *term frequency-inverse document frequency*) (22) foi utilizado durante a etapa de pré-processamento para representação dos documentos por favorecer termos significativos que aparecem poucas vezes no conjunto de documentos e desfavorecer termos que aparecem com bastante frequência durante a análise. Este beneficiamento dos termos é obtido através da Equação 1.

$$idf(T_j) = \log \frac{M}{DF_j} \tag{1}$$

sendo que M é a quantidade de documentos presentes no conjunto de dados e DF_j é o número de documentos em que o termo T_j ocorre. Combinando o fator de beneficiamento com a medida *tf*, obtemos a Equação 2:

$$tfidf(T_j) = TF_j \cdot \log \frac{M}{DF_j} \tag{2}$$

O valor de TF_j é dado pela frequência dos termos (do inglês *term frequency* (*tf*)),

ou seja, o atributo $a_{ij} = tf(T_j, documento_i)$. No qual $tf(T_j, documento_i)$ é o valor da quantidade de ocorrências do termo T_j no $documento_i$.

Ao mesmo tempo que foram calculados os valores *TF-IDF* do conjunto de dados, também foram removidas as palavras típicas da língua conhecidas como *stopwords* (29) e aplicadas radicalização de palavras PorterStemming (38) que transformam as formas derivadas de uma palavra a um único radical removendo os sufixos e afixos. As *stopwords* são termos comuns utilizadas na comunicação e escrita de qualquer idioma, como por exemplo na língua inglesa: *a, an, all, be, but*, entre outros. As *stopwords* afetam o processo de recuperação de informação e ponderação *TF-IDF*, pois possuem alta frequência e pouco significado (28). Em relação a radicalização, sua aplicação garante que as variações de uma palavra sejam eliminadas da representação *TF-IDF*, por exemplo, as palavras *opening*, *opened* e *opener* seriam substituídas por *open*. Esta técnica reduz a quantidade de termos a serem analisados o que pode facilitar o processo de agrupamento de dados e recuperação de informação relevante.

Quando se trata de conjuntos de dados cuja origem são textos, o processo de análise se torna mais complexo, uma vez que os dados foram transformados para tipos numéricos e o conjunto resultante do pré-processamento é de natureza altamente esparsa.

Os procedimentos realizados para representação dos conjuntos de dados no modelo *Bag-of-Words* e remoção de *stopwords* foram executados utilizando as bibliotecas do projeto *Apache Mahout*[3]. Este projeto provê algoritmos para o aprendizado de máquina (24) desenvolvidos sobre o modelo de programação *MapReduce*.

As principais técnicas utilizadas para o desenvolvimento deste trabalho serão apresentadas nas próximas seções.

3.1 *MapReduce*

O *MapReduce*(3) é um modelo de programação para processamento de dados de forma paralela e distribuída. O paradigma *MapReduce* possui duas fases: *Map* e *Reduce*. Cada fase necessita de um par <*key,value*> (4). A fase *Map* é responsável pelo mapeamento dos dados de entrada, enquando a fase *Reduce* fica encarregada pela união dos resultados dos pares <*key,value*>

A Figura 1 mostra um exemplo do funcionamento do algoritmo para contagem de palavras utilizando o modelo *MapReduce*.

Na fase *Map*, arquivo de entrada é divido e mapeado de acordo com a quantidade de linhas nele/do documento, cada palavra é separada e atribuído a um par <*key,value*>. As palavras são *keys* e suas frequências nas linha são os *values*. Existe uma etapa intermediária

[3]Projeto *Apache Mahout*^{TM} disponível em: <http://mahout.apache.org/>

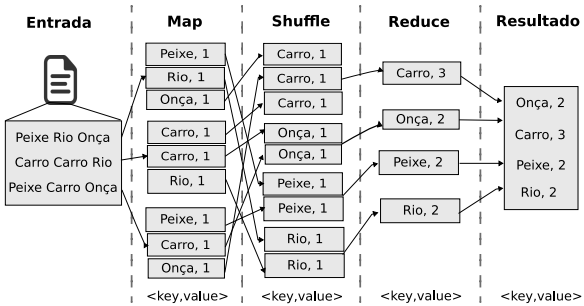


Figura 1. Exemplo para contagem do número de palavras de um documento no modelo de programação *MapReduce*.

as fases *Map* e *Reduce*, denominada de *Shuffle*. Nesta etapas os dados são embaralhados e ordenados de acordo com as respectivas *keys*. A fase *Reduce* é responsável por agregar pares <*key,value*> que contenham as mesmas *keys*. A fase *Reduce* no exemplo *WordCount*, une todas as *keys* idênticas e soma os valores presentes nos *values* emitidos na fase *Map*. O resultado final é o número de ocorrências de determinada palavras no documento.

Todas estas etapas são distribuídas pelo *cluster* de computadores para serem processadas em paralelo. Este modelo permite utilizar algoritmos em uma máquina ou milhares de máquinas e utilizados nas abordagens proposta neste trabalho.

3.2 *K-means*

A análise dos dados pode ser feita utilizando diversos algoritmos da área, além de ser uma método útil na busca de padrões relevantes aos conjuntos de dados desconhecidos. A técnica de agrupamento *K-means* (39) é reconhecida por sua simplicidade e pela aplicação em diversas áreas, além de ser um dos procedimentos mais conceituados da área de mineração de dados que possuí implementação no modelo *MapReduce* (6, 40). A descrição do *K-means* é apresentada no Algoritmo 1.

A Figura 2 apresenta o algoritmo *K-means* desenvolvido sobre o modelo de programação *MapReduce* (41) disponível na biblioteca *Mahout* (24) utilizada no desenvolvimento deste trabalho.

Durante a execução do algoritmo o conjunto de dados é divido em blocos e são escolhidos k centroides iniciais escolhidos aleatoriamente. Os blocos são encaminhados a função *Map* que realiza o cálculo da similaridade de cada um dos objetos aos k centroides, em seguida os objetos são atribuídos ao centroide mais próximo, ou seja, o centroide que possui

Algorithm 1 Algoritmo K-means

Dado conjunto de dados D, k o número de grupos e t o número máximo de iterações

Require: $k \geq 2$ e $t \geq 1$
Ensure: C centroides
1: Inicializa k centroides C de forma aleatória;
2: **repeat**
3: **for all** Objeto x_i do conjunto de dados D **do**
4: **for all** Centroide/protótipois iniciais **do**
5: Adiciona o objeto x_i ao grupo C_k com a menor similaridade $dist = \| x_i - C_k \|^2$
6: **end for**
7: **end for**
8: **for all** C_k em Centroide/Grupo **do**
9: Recalcular o novo centroide $C_k = média(\text{objetos em } C_k)$
10: **end for**
11: **until** Atingir convergência ou o número máximo de iterações / Centroides não se alterem

características similares ao do objeto comparado. Como todo programa desenvolvido no modelo *MapReduce*, esta fase utiliza pares <*key,value*>. Em seguida, os centroides são encaminhados para a função *Reduce*, onde são recalculados os centroides com base nos objetos atribuídos a cada um. Caso o valor dos centroides sejam alterados, seus valores são atualizados e o processo repetido até que os centroides não se alterem mais.

O método de agrupamento *K-means* pode ser aplicado à diversos tipos de dados, no entanto o seu desempenho é influenciado por alguns fatores: (i) A escolha do número *k* de centroides iniciais, que serão selecionados de forma aleatória; (ii) A sensibilidade a inicialização, ficando estagnado em mínimos locais (40, 42).

Para evitar as deficiências do algoritmo podemos aplicar diversas alternativas durante o pré-processamento investindo a obtenção de uma boa representação do conjunto de dados para que o procedimento de agrupamento tenha bons resultados. A aplicação de procedimentos para redução de dimensionalidade e seleção de atributos tem como objetivo garantir que os dados sejam representados de forma que o desempenho do algoritmo *K-means* não seja influenciado pela alta dimensionalidade e/ou representação inadequada dos conjuntos de dados.

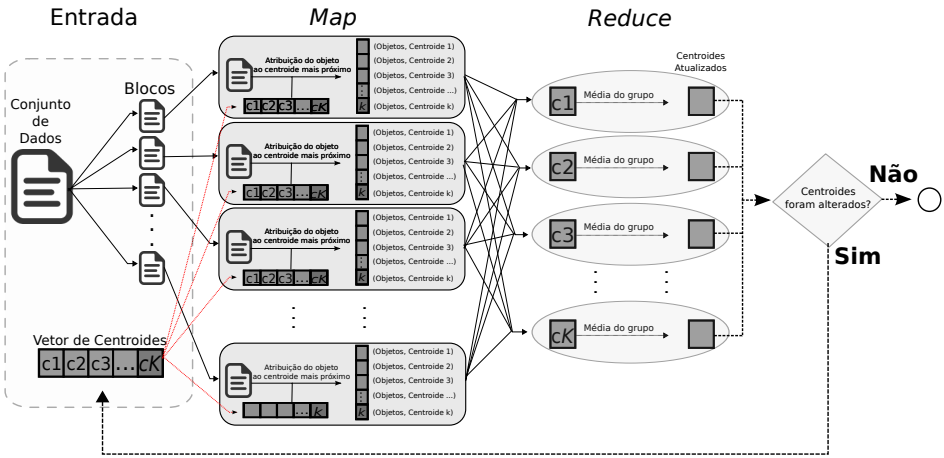


Figura 2. Execução do algoritmo *K-means* da biblioteca Mahout no modelo de programação *MapReduce*.

3.3 Análise de Componentes Principais

Análise de Componentes Principais (8) (ACP) é um dos procedimentos utilizados para reduzir as dimensões de um conjunto de dados de acordo com padrões encontrados durante a análise automática dos principais componentes dos dados. Ao analisar grandes conjuntos de dados, é imprescindível a redução de sua dimensão para facilitar a manipulação e reduzir o tempo gasto para processar este grande volume de dados.

O mesmo procedimento é utilizado para análise de dados textuais em razão de serem muito esparsos o que prejudicam as medidas de similaridades espaciais como a distância euclidiana, utilizada no algoritmo *K-means*.

A técnica consiste em substituir p variáveis de um conjunto de dados por um número reduzido de variáveis q, que é a representação dos principais componentes do aglomerado de dados. Além disso, sua representação em formatos reduzidos é produzida identificando padrões existentes nos dados, conservando as semelhanças e diferenças entre os objetos do conjunto de dados (8).

No procedimento ACP é necessário calcular a matriz de covariância dos dados de entrada operação usada para encontrar a relação entre as dimensões do conjunto de dados. Em seguida, a base do vetor para resolução do método é calculada encontrando os autovalores (do inglês, *eigenvalues*) e autovetores(do inglês, *eigenvector*) por meio da equação $R^T(XX^T)R = \Lambda$, em que X é a matriz de dados, R é uma matriz de autovetores e Λ cor-

responde a matriz diagonal de autovalores. A projeção de um determinado dado $C_n = R_N^T X$ da dimensão do espaço original p para um subespaço gerado por n principais componentes que minimizam o erro médio quadrático (9).

O algoritmo ACP é executado com 3 etapas *MapReduce*, como é descrito no Algoritmo 2.

Algorithm 2 Algoritmo Análise de Componentes Principais (MapReduce)

Dado n o número de dimensões para o novo conjunto de dados, ou seja, o número de componentes principais que serão utilizados para reduzir o conjunto de dados original.

Require: Matriz X_{kl} ($k =$ número de documentos e $l =$ número de palavras dos conjuntos de dados) e $n \geq 2$

Ensure: Matriz reduzida X'_{kn}

1: Calculo da média dos valores de cada coluna da matriz X em *MapReduce*;
2: Subtração da média obtida da matriz original X em *MapReduce*;
3: Calculo da matriz de covariância, autovalores e autovetores em *MapReduce*;
4: Escolha dos n maiores autovalores e autovetores;

Deste modo, são peças principais durante a realização deste procedimento de redução de dimensionalidade os autovalores e autovetores. Autovetores que possuem os maiores autovalores são componentes principais dos conjuntos de dados, portanto são utilizados como variáveis na formação do novos conjuntos de dados reduzidos.

3.4 *Latent Dirichlet Allocation*

Latent Dirichlet Allocation (10) é uma metologia probabilística que descreve um corpo textual através de tópicos, aplicado a diversas áreas para recuperação de informação (43).

A inferência estatística utilizada pelo algoritmo LDA é descrita de forma que cada tópico possui uma distribuição de probabilidade de acordo com as palavras contidas em cada documento (44). A escolha de B tópicos define a probabilidade da *i*-ésima palavra de um documento como:

$$P(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j) \tag{3}$$

no qual z_i é uma variável oculta (do inglês, *latent variable*) e indica o tópico que a íesima palavra vai integrar. Em seguida, $P(w_i|z_i = j)$ determina a probabilidade de uma palavra w_i

participar do j-ésimo tópico. $P(z_i = j)$ define a probabilidade de escolha de uma palavra w_i do tópico j no documento atual, sua variação é dada através de diferentes documentos.

Através de $P(w|z)$ é definido a importância de uma palavra para um determinado tópico, uma vez que $P(z)$ estabelece a probabilidade de vínculo de um tópico no corpo textual do documento. Sendo assim, cada documento é descrito com uma participação de um conjunto distinto de termos. Ao visualizar o corpo textual como uma mistura de probabilidades, podemos descrever $P(w|z)$ com um conjunto de D distribuição multinomial (devido a possibilidade de existência de dois resultados de interesse) ϕ sobre as palavras W. Deste modo $P(w|z)$ é descrito pela Equação 4.

$$P(w|z = j) = \phi_w^{(j)} \tag{4}$$

Portanto $P(z)$ é denotado com um conjunto de D distribuição multinomial θ de acordo com B tópicos tópicos a serem descobertos. Para uma palavra em um documento d, $P(z = j) = \theta_j^{(d)}$. A técnica LDA combina a Equação 3 com a distribuição de probabilidade em θ provendo um modelo generativo completo para documentos (44, 45).

O Algoritmo 3 apresenta a sequência de operações realizadas pelo algoritmo LDA desenvolvido no modelo *MapReduce*, para sua execução são necessárias definições de alguns parâmetros de entrada. Os autores utilizaram como entrada a representação TF-IDF dos conjuntos de dados, devido as características apresentadas na Seção 3, definiram a quantidade de tópicos a serem descobertos de acordo com as quantidades de grupos existentes em cada um dos conjuntos de dados, limitaram as iterações do algoritmo em 20 vezes e utilizaram valores *default* para os demais parâmetros.

Algorithm 3 Algoritmo *Latent Dirichlet Allocation*

Require: Conjunto de dados na representação TF ou TF-IDF, definição do número B de tópicos, definir número de iterações i e outros parâmetros (utilizados com valor *default* do ambiente).

Ensure: B de tópicos, probabilidade de cada documento D pertencer a cada B tópicos

1: Iniciar B tópicos de forma aleatória
2: **repeat**
3: Executar tarefas *Map* (Veja o Algoritmo 4);
4: Execute tarefa *shuffle*;
5: Executar tarefas *Reduce* (Veja o Algoritmo 5), ordenando e unindo as mudanças referentes ao número de tópicos encontrados;
6: **until** algoritmo convergir **or** número de iterações $>= i$
7: **return** B tópicos com w palavras e Probabilidade de cada um dos B tópicos pertencer a cada documento;

Cada função *Map* (Algoritmo 4) realiza os cálculos de probabilidade (Equação 3) para cada um das w palavras de cada um dos documento dos conjuntos de dados em paralelo, retornando as palavras que constituem os B tópicos a serem encontrados pelo algoritmo.

Algorithm 4 Função *Map* do algoritmo *Latent Dirichlet Allocation*

1: **for all** documentos D **do**
2: **for all** palavras w **do**
3: calcular probabilidade da palavra do documento pertencer a um dos tópicos $P(w_i) = \sum_{j=1}^{T} P(w_i|z_i = j)P(z_i = j)$
$P(w|z = j) = \phi_w^{(j)}$
4: **end for**
5: **end for**
6: **return** novo tópicos com w palavras para cada documento

Em seguida, essas informações são passadas a função *Reduce* (Algoritmo 5) que é responsável pela união das palavras pertencentes ao mesmo tópico e também atualizar os valores que definem as probabilidades de participação de cada um dos tópicos nos documentos

Algorithm 5 Função *Reduce* do algoritmo *Latent Dirichlet Allocation*

1: **for all** B Tópico descoberto **do**
2: Atualizar as palavras presentes em cada tópico
3: Definir a porcentagem de participação cada tópico para os documentos
4: **end for**
5: **return** B tópicos com w palavras;

A saída do algoritmo LDA resulta em dois arquivos, no primeiro são apresentados os termos que cada um dos B tópicos contém, no segundo os valores de probabilidade que cada tópico descoberto tem na caracterização do documento.

A seguir, serão apresentadas as 4 abordagens avaliadas neste trabalho.

3.5 Abordagens utilizadas

Nesta Seção serão apresentados as estratégias realizadas para avaliação do agrupamento de dados sobre diferentes conjuntos de dados textuais.

O valor fixado para redução de dimensionalidade com o algoritmo ACP foi escolhido empiricamente, e limitada pelos *hardwares commodities* disponíveis para o desenvolvimento deste trabalho. Para a abordagem de seleção de atributos proposta nesse trabalho, foram feitas

análises da seleção de 10, 50, 100, 250 e 500 atributos/termos mais relevantes da representação do algoritmo LDA em tópicos.

3.5.1 Abordagem *TF-IDF* Nesta estratégia, os conjuntos de dados textuais foram representados no modelo VSM pelo fator de peso *TF-IDF*, no qual foram aplicados *stopwords* para remoção de palavras com alta frequência e aplicadas as radicalizações *PorterStemming* para redução das palavras derivadas de um mesmo radical. Logo em seguida, o algoritmo *K-means* foi aplicado de acordo com os números de grupos esperados para cada um dos conjuntos.

3.5.2 Abordagem ACP e K-means Durante esta estratégia, as representações *TF-IDF* dos conjuntos de dados foram reduzida para 100 dimensões utilizando o algoritmo ACP para extração dos atributos, ou seja, os 100 termos mais relevantes foram consideradas para formarem os novos conjuntos de dados. Após a redução de dimensionalidade foi realizado o agrupamento do conjunto de dados utilizando o algoritmo *K-means* de acordo com os números de grupos esperados para cada um dos conjuntos.

3.5.3 Abordagem LDA Nesta estratégia, avaliamos a precisão do algoritmo LDA para o agrupamento e extração de atributos para B tópicos relacionados ao número de grupos correto esperado para cada um dos conjuntos de dados.

Não foi aplicado algoritmo *K-means* para o agrupamento, pois o algoritmo LDA representa os documentos dos conjuntos de dados em relação a probabilidade do documento pertencer a um dos tópicos descobertos, formato utilizado para representação de grupos e verificar acurácia dos tópicos obtidos.

3.5.4 Abordagem LDA e Seleção de atributos Nesta etapa, foram escolhidos os melhores resultados obtidos durante a etapa descrita na Seção 3.4 para realização da seleção de atributos/termos. Por exemplo, ao escolher 10 melhores termos de cada um dos B tópicos desejados, ou seja, serão selecionados os 10 termos com as maiores probabilidades para cada um dos B tópicos encontrados, totalizando 50 características relevantes para distinção dos documentos. O objetivo desta seleção foi separar os diferentes assuntos tratados em cada um dos conjuntos de dados pelos principais termos que parte dos tópicos descobertos/encontrados pelo algoritmo LDA.

Após a seleção de atributos, os conjuntos de dados sofreram transformações em suas características, eles passaram a serem compostos por apenas 50 termos ou dimensões. Os documentos que não apresentavam nenhum dos 50 termos selecionados foram removidos, em seguida os conjuntos de dados foram encaminhados ao algoritmo *K-means*.

A visão geral das abordagens utilizadas é apresentado na Figura 3

4 Experimentos

Os experimentos foram executados em um cluster *Hadoop* 2.4.0 com integração *Mahout* constituído por 5 computadores *commodities* com processador Intel Core i5 de 2.5 GHz, sistema operacional Linux, memória RAM de 4 GB e disco rígido de 500 GB, todas as máquinas estavam conectadas por uma rede local, deste modo não há interferência no fluxo de dados da rede durante a execução dos experimentos.

A fim de avaliar a acurácia da metodologia foram utilizados diferentes conjuntos de dados reais, apresentados na Seção 4.1.

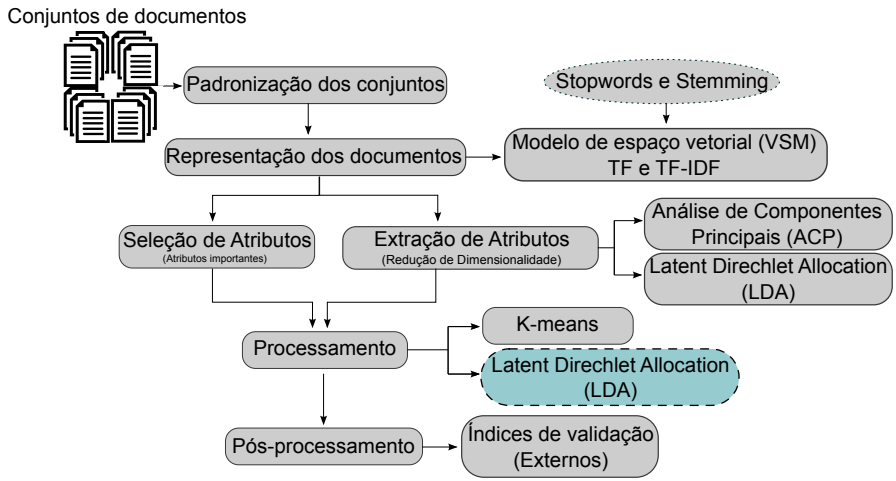


Figura 3. Visão geral da metodologia aplicada.

4.1 Conjunto de dados

Para a execução dos experimentos os autores elaboraram 4 conjuntos de dados textual obtidos através de *Medline PubMed*[4] (46), os artigos foram selecionados com base nos conceitos: *Chemical*, *Disease*, *Gene*, *Mutation* e *Species*, de acordo com a ferramenta PubTator (47). Foi elaborado 1 conjunto com base no repositório *20Newsgroup*[5] denominados

[4]Disponível em:http://www.ncbi.nlm.nih.gov/pubmed
[5]Repositório *20Newsgroups* disponível em: http://qwone.com/ jason/20Newsgroups/

20News.Dif5. Além disso foi utilizado os conjuntos *SMART*[6] e *WebK*[7]. Todos os conjuntos de dados se diferenciam pela quantidade de objetos a serem processados e também pelos assuntos tratados em cada documento de grupo distinto. Os dados textuais são de natureza altamente esparsa e possuem grande/alta dimensionalidade (24).

A Tabela 1 apresenta de forma resumida a quantidade de objetos em cada um dos conjuntos de dados, o número correto de grupos, o número de atributos da representação *TF-IDF*, a quantidade de atributos considerados para a redução de dimensionalidade ACP e o numero de atributos utilizados para a seleção de atributos conforme os resultados do algoritmo LDA.

Tabela 1. Características dos conjuntos de dados

Conjunto de Dados	# grupos	# objetos	# atributos TF-IDF	# atributos ACP	# atributos Seleção Atributos
PubMed Amostra	5	$5 \cdot 10^3$	29809	100	50
PubMed1	5	$5 \cdot 10^5$	491057		
PubMed2	5	10^6	801770		
PubMed3	5	$2.5 \cdot 10^6$	1020080		
20news.Dif5	5	$4.686 \cdot 10^3$	19273		
SMART	4	$7.095 \cdot 10^3$	13859		
WebK	4	$4.202 \cdot 10^3$	15039		

A quantidade de atributos na representação *TF-IDF* (Tabela 1) aumenta de acordo com as quantidades de dados dos conjuntos de dados, mas também sofre influência da quantidade de elementos presentes em seu corpo textual. O corpo textual dos conjuntos de dados PubMed são compostos pelo título do artigo, nome dos autores, resumo do artigo (*abstract*). Considerando o corpo textual do conjunto de dados *SMART* são compostos apenas por título dos documentos. Observando o conjunto de dados *20news.Dif5*, sua organização é feita apenas por mensagens de textos sobre 5 assuntos diferentes. Os documentos que estruturam o conjunto de dados *WebK* são páginas *web* definidas com HTML e suas *tags* de marcação. Todas essas características influenciam a representação dos documentos e sua dimensionalidade.

4.2 Avaliação de Resultados

Os resultados de agrupamentos obtidos foram avaliados de acordo com os índices externos *Ajusted Rand* (AR), *Fowlkes and Mallows* (FM) e *Mirkins* (M) (48). Os índices externos avaliam a similaridade entre agrupamentos de dados, desta forma podemos verificar

[6]Repositório *SMART* disponível em: http://www.dataminingresearch.com/index.php/ 2010/09/classic3-classic4-datasets/
[7]Repositório *WebK* disponível em: http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/

a semelhança entre os resultados obtidos após o agrupamento de dados com os resultados "reais" ou esperados.

Considerando cada par de objetos x_i e x_j pertencente ao conjunto de dado X, tal que $i \neq j$, os grupos $C_r \in \pi_r$ e $C_g \in \pi_g$. Os agrupamentos podem ser classificados da seguinte forma:

- **11**: $(x_i \in C_r)$ e $(x_j \in C_r)$ em π_r,$(x_i \in C_g)$ e $(x_j \in C_g)$ em π_g
- **01**: $(x_i \in C_r)$ e $(x_j \in C_r)$ em π_r,$(x_i \notin C_g)$ e $(x_j \in C_g)$ em π_g
- **10**: $(x_i \in C_r)$ e $(x_j \in C_r)$ em π_r,$(x_i \in C_g)$ e $(x_j \notin C_g)$ em π_g
- **00**: $(x_i \in C_r)$ e $(x_j \notin C_r)$ em π_r,$(x_i \notin C_g)$ e $(x_j \in C_g)$ em π_g

Seja np_t o número de todas as possíveis combinações de pares de objetos do conjunto de dados ($np_t = n_o(n_o - 1)/2$), sendo n_o a quantidade total de objetos, np_{11} o número de pares de objetos na situação **11**, np_{01} o número de pares de objetos na situação **01**, np_{10} e np_{00} os números de pares de objetos nas situações **10** e **00**, respectivamente.

O índice Rand Ajustado (*Ajusted Rand*) é apresentado na Equação 5. O Índice *Ajusted Rand* considera apenas os pares de objetos que estão no mesmo agrupamento, ou seja, os objetos serão semelhantes quando estiverem no mesmo grupo, caso contrario, são considerados distintos.

$$AR(\pi_a, \pi_b) = \frac{np_{11} - \frac{(np_{11}+np_{01})(np_{11}+np_{10})}{np_t}}{\frac{(np_{11}+np_{01})+(np_{11}+np_{10})}{2} - \frac{(np_{11}+np_{01})(np_{11}+np_{10})}{np_t}} \tag{5}$$

Algumas variações surgiram a partir do índice textitRand — RI (48) também foram aplicadas, tais como o Índice *Fowlkes and Mallows* e o Índice *Mirkin's*. O Índice FM é mostrado na Equação 6. Os seus valores são caracterizados próximos de 1 quando os agrupamentos são correspondentes, caso contrario assume valores perto de 0.

$$FM(\pi^a, \pi^b) = \frac{np_{11}}{\sqrt{(np_{11} + np_{10})(np_{11} + np_{01})}} \tag{6}$$

E, por último, o Índice *Mirkin's* mostrado na Equação 7. o MI é um ajuste do Índice Rand (49) e assume o valor 0 quando os agrupamentos comparados são idênticas, caso contrário assume um valor $M \in \mathbb{N} \mid M \geqslant 1$.

$$M(\pi^a, \pi^b) = n_o(n_o - 1)(1 - R(\pi^a, \pi^b)) \tag{7}$$

A utilização dos índices de validação externos contribuem para verificar a porcentagem de acertos obtida após a realização do agrupamento sobre os dados originais e pré-processados.

Tabela 2. Resultado da avaliação de agrupamento realizado sem redução de dimensionalidade, com a redução de dimensionalidade (ACP e LDA) e após a seleção de atributos

Base de dados	AR	FM	M
20news.Dif5	0,5141 (0,1156)	0,6231 (0,0867)	3683998 (966354,5984)
20news.Dif5 ACP	0,1695 (0,04770)	0,3409 (0,0387)	5974983,4 (334067,4648)
20news.Dif5 LDA	**0,6215 (0,0950)**	**0,7017 (0,0729)**	**2757987,8 (749216,1655)**
20news.Dif5 S.A	0,2307 (0,0290)	0,4487 (0,01796)	6375036,6 (458566,8928)
SMART	0,3668 (0,0833)	0,5512 (0,0589)	13146286,6 (1762148,9294)
SMART ACP	0,2197 (0,0622)	0,4452 (0,0446)	16148784,4 (1313971,2703)
SMART LDA	**0,5585 (0,1089)**	**0,6866 (0,0787)**	**9139552,2 (2207734,4200)**
SMART S.A	0,1103 (0,0181)	0,4237 (0,0161)	18005891,4 (1238797,2884)
WebK	0,1275 (0,0557)	0,3777 (0,0361)	6303312,2222 (509122,9369)
WebK ACP	0,1050 (0,0523)	0,3517 (0,0393)	6316344 (353189,3695)
WebK LDA	**0,3870 (0,1519)**	**0,5654 (0,1080)**	**6979328 (2836416,6249)**
WebK S.A	0,0392 (0,0507)	0,3237 (0,03903)	6904330 (370633,4718)
Pubmed Amostra	0,6150(0,0915)	0,67218985(0,0998)	3481792,2 (1117456,0871)
Pubmed ACP	0,2161(0,0331)	0,3782(0,0257)	6538960,4(302513,0430)
Pubmed Amostra LDA	0,4580 (0,0656)	0,5710 (0,0518)	4538255,8000 (574423,1100)
Pubmed Amostra S.A	**0,9129 (0,0035)**	**0,9303641784 (0,0028)**	**1519219,8000 (1710553,7768)**
Pubmed1	0,2691 (0,0719)	0,2534 (0,0461)	58047624965,25 (2810098288,6933)
Pubmed1 ACP	0,0633 (0,0094)	0,3083 (0,0042)	77091362609,6 (1065096644,6425)
Pubmed1 LDA	0,2084 (0,0295)	0,3784 (0,0178)	66796137381(4973998978,0291)
Pubmed1 S.A	**0,3307 (0,0557)**	**0,4912 (0,0607)**	**57982706720 (4904625643,4351)**
Pubmed2	0,2504 (0,0396)	0,2508 (0,0227)	250748977076,4 (22734240895,4274)
Pubmed2 ACP	N/D	N/D	N/D
Pubmed2 LDA	0,0989 (0,0165)	0,2820 (0,0128)	291747274583,2 (6554022147,7149)
Pubmed2 S.A	**0,5900 (0,0255)**	**0,6726 (0,0203)**	**154679709596,4 (46338658563,6303)**
Pubmed3	0,2289 (0,0736)	0,3904 (0,0581)	2421646975518,86 (1804483360022,48)
Pubmed3 ACP	N/D	N/D	N/D
Pubmed3 LDA	0,1639 (0,0862)	0,3334 (0,0700)	1689257500000 (164878101752,813)
Pubmed3 S.A	**0,3068 (0,0997)**	**0,4643 (0,0666)**	**1480211294799 (283082467747,628)**

4.3 Análise dos Resultados

A Tabela 2 mostra a média e o desvio padrão dos resultados de 10 execuções alcançado pela avaliação de agrupamento dos conjuntos de dados descritos de acordo com o número conhecido de grupos apresentados na Seção 4.1 e os índices de avaliação externos mostrados na Seção 4.2. Os melhores resultados das avaliações são apontados em negrito separadamente pelos índices externos, dois conjuntos de dados não foram avaliados para o algoritmo de

redução de dimensionalidade ACP devido os conjuntos serem altamente esparsos e exigirem configurações de hardware que superam os limites dos *hardwares commodities* utilizados no desenvolvimento do trabalho.

Levando em consideração os conjuntos de dados que possuem até 10^3 objetos, os melhores resultados foram obtidos com aplicação apenas do algoritmo LDA (3) que estão em negrito, no qual o algoritmo foi executado os T tópicos especificados de acordo com o número de grupos presente na Tabela 1. Considerando os conjuntos de dados com mais de 10^5 objetos, os melhores resultados foram obtidos com a seleção de atributos do algoritmo LDA (4).

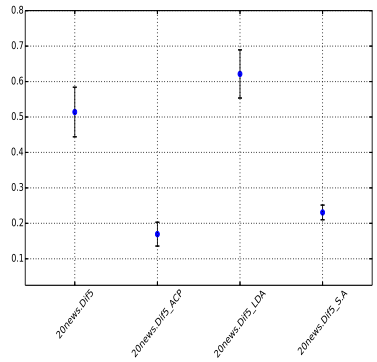
Os resultados obtidos apenas com a utilização do algoritmo *K-means* (1) para o agrupamento foram melhores que todas as tentativas de redução de dimensionalidade com o método ACP. Os resultados experimentais mostram que a aplicação do algoritmo ACP a grandes conjuntos de dados avaliados não favorece a realização da etapa de agrupamento, isso devido a péssima representação obtida pelo método para conjuntos com grande dimensionalidade de dados. Deste modo, os resultados avaliados são piores que os obtidos com o agrupamento partir do conjunto de dados da abordagem *TF-IDF* (1).

Por outro lado, a abordagem (3) LDA que utiliza o algoritmo LDA obteve melhores resultados quando aplicado aos conjuntos de dados que possuem até 10^4 documentos. Ao lidar com conjuntos textuais maiores que 10^5 documentos a acurácia da abordagem LDA (4) diminui em razão da quantidade de palavras, ou seja, a dimensionalidade dos conjuntos.

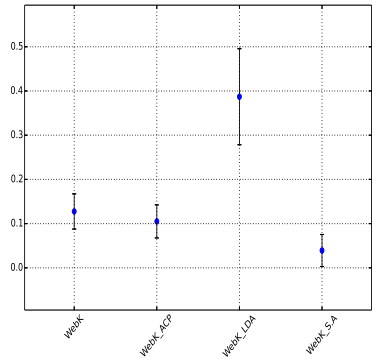
A abordagem (4) LDA com a seleção de atributos desenvolvida baseado nos termos presentes em cada tópico encontrado pelo algoritmo LDA é uma abordagem que traz uma melhora de $\approx 13\%$ para o processo de agrupamento de documentos textuais dos conjuntos de dados com proporções de um contexto de dados escalável. Pois, a representação dos conjuntos de dados é feita pela seleção de atributos significativos inferidos estatisticamente, garantindo a escolha de termos relevantes para os conjuntos de dados. Deste modo, a representação dos dados textuais deixam de ser altamente esparsos mantendo características relevantes para a etapa de agrupamento de dados.

Os resultados da média e desvio padrão do índice Rand Ajustado de 10 execuções foram utilizados para análise utilizando a distribuçao T-Student. Os autores assumiram um Intervalo de Confiança de 95%, obtendo um valor de α de 0.05. Em seguida, foram calculados os valores de t para cada abordagem dado por distribuição $t = t_{[1-\alpha/2;n-1]} \times \frac{DP}{\sqrt{n}}$, onde n é definido pelo número de amostras, DP é o desvio padrão da amostra, $n-1$ representa graus de liberdade (50, 51).

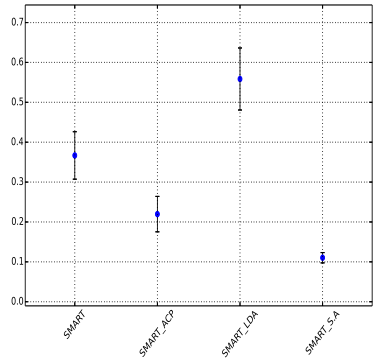
As Figuras 4(a),4(b),4(c),4(d),5(a),5(b),5(c) mostram os gráficos de comparação entre as abordagens propostas nesse trabalho definidos pela media $\pm$ Intervalo de Confiança. Foram feitos testes visuais levando em conta a sobreposição dos intervalos de confianças de



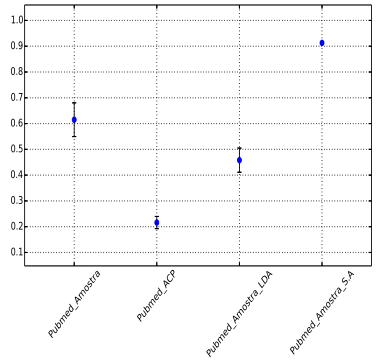
(a) Análise da média do conjunto de dados 20news.diff5 para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x.



(b) Análise da média do conjunto de dados WebK para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x.



(c) Análise da média do conjunto de dados SMART para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x.

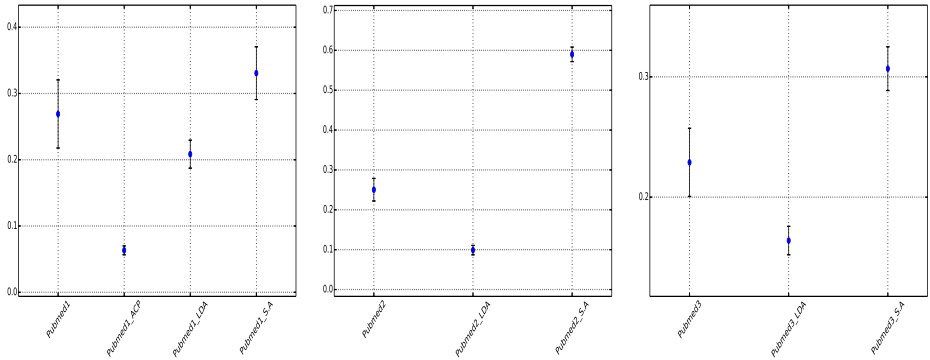


(d) Análise da média do conjunto de dados PubMed Amostra para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x.

Figura 4. Avaliaçao das abordagens dos conjunto 20news.Diff5, WebK, SMART e PubMed Amostra por meio média $\pm$ Intervalo de Confiança

cada abordagem.

No conjunto de dados 20news.Diff5 (Ver Figura4(a)), temos sobreposição entre as abordagens TF-IDF e LDA. Abordagem LDA é melhor que as abordagens ACP e S.A. Abordagem TF-IDF melhor que ACP e S.A. Não há sobreposição de intervalos de confianças.



(a) Análise da média do conjunto de dados PubMed1 para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x.

(b) Análise da média do conjunto de dados PubMed2 para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x. Não foram apresentados valores para abordagem ACP

(c) Análise da média do conjunto de dados PubMed3 para as abordagens propostas neste trabalho. Valor de AR no eixo y e bases no eixo x. Não foram apresentados valores para abordagem ACP

Figura 5. Gráficos da avaliaçao das abordagens dos conjunto PubMed1, PubMed2 e PubMed3 por meio média $\pm$ Intervalo de Confiança

No conjunto de dados WebK (Ver Figura4(b)). Abordagem LDA é melhor que as abordagens TF-IDF, ACP e S.A. Entretanto as abordagens TF-IDF, ACP e S.A tem sobreposição de intervalos de confiança, requerendo aplicação do teste-t.

No conjunto de dados SMART (Ver Figura4(d)). Abordagem LDA é melhor que as abordagens TF-IDF, ACP e S.A. Abordagem TF-IDF melhor que ACP e S.A. Não há sobreposição de intervalos de confianças.

No conjunto de dados PubMed Amostra (Ver Figura4(b)). Abordagem S.A é melhor que as abordagens TF-IDF, ACP e LDA. Abordagem TF-IDF melhor que ACP e LDA. Por fim abordagem LDA melhor que ACP. Não há sobreposição de intervalos de confianças.

No conjunto de dados PubMed1 (Ver Figura5(a)), temos sobreposição entre algumas abordagens. Sendo necessário aumentar o número de replicações ou aplicação de do teste-t. Entretanto, a abordagem S.A com 50 atributos é melhor que as abordagens LDA e ACP. Logo a abordagem LDA é melhor que a abordagem ACP, mas não é melhor que a abordagem TF-IDF.

No conjunto de dados PubMed2(Ver Figura 5(b)) e PubMed3 (Ver Figura 5(c)), todas as abordagens são diferentes entre si, sendo a abordagem S.A com 50 atributos a melhor

devido o seu valor da média ser maior.

A avaliação da seleção de atributos proposta foi elaborada utilizando 10, 50, 100, 250 e 500 termos mais relevantes dos tópicos encontrados pelo algoritmo LDA. A Figura 6 apresenta a média de 10 execuções do algoritmo *K-means* sobre o conjunto de dados obtido após a seleção de atributos. Os valores apresentados são referentes ao índice AR que considera apenas pares de objetos que estão no mesmo agrupamento de dados.

Ao observar os conjuntos textuais pequenos (*20news.dif5*, *WebK* e *SMART*), quanto maior a quantidade de termos considerados durante a seleção, melhor são os valores obtidos pelo índice AR. Já os conjuntos textuais grandes (*PubMed Amostra*, *PubMed1*,*PubMed2* e *PubMed3*) essa relação é inversa. Quanto maior a quantidade de termos a serem consideradas para elaboração do novo espaço dimensional, pior os resultados obtidos.

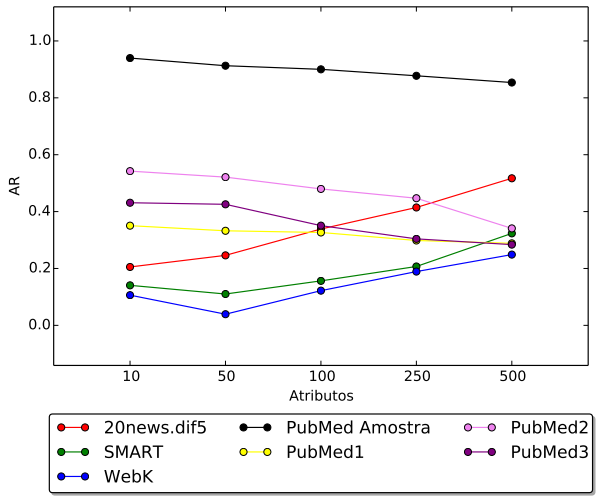


Figura 6. Relação AR vs Quantidade de Atributos utilizados na seleção de atributos para geração da nova representação dos conjuntos de dados .

As características das bases construídas pelos autores devem ser consideradas durante a avaliação. Os assuntos tratados por elas possuem relações que ocasionam sobreposição de classes, influenciando o agrupamento.

5 Conclusão

Os resultados apresentados na Tabela 2 permitem concluir que as técnicas para redução de dimensionalidade (ACP e LDA) quando aplicadas a grandes conjuntos de dados textuais não favorecem o agrupamento. Sendo assim, a técnica de agrupamento de dados *K-means* obtém melhores resultados sobre grandes conjuntos de dados quando não realizamos a redução de dimensionalidade. Tais resultados foram verificados em diferentes conjuntos de dados propostos neste trabalho.

O método para seleção de atributos desenvolvido e avaliado pelos autores, obtém melhores resultados quando aplicada a conjuntos de dados textuais maiores utilizando apenas 50 atributos/termos para representação de cada documentos. Em contrapartida, a utilização método LDA obtém melhores resultados quando aplicado a conjuntos textuais que possuem poucos documentos.

As abordagens estudadas possuem implementações sobre o modelo MapReduce, o que possibilita a utilização do processamento paralelo e distribuído de forma escalável, podendo aumentar ou diminuir o número de máquinas a serem utilizadas de acordo com as necessidade de processamento.

Com o crescimento da quantidade de dados geradas atualmente, a comparação de técnicas e algoritmos desenvolvidos devem ser aplicadas em conjuntos de dados reais mais robustos para que os resultados obtidos tenham sejam validos para utilização no campo da pesquisas e na sociedade.

Como trabalhos futuros, devem ser avaliados outros métodos para representação *Bag-of-Words*, que proporcionem a ponderação de termos melhor aos conjuntos de dados e também avaliação de outras técnicas para pré-processamento de dados. Também será investigado a utilização do uso de *meta-feature* para representação das característica dos documentos como um conjunto.

6 Agradecimentos

Os autores agradecem a Universidade Federal de Viçosa — *Campus* Rio Paranaíba (UFV-CRP), pela disponibilidade do laboratório de pesquisa do curso de Sistemas de Informação utilizado no desenvolvimento deste projeto, ao CNPq e FAPEMIG por financiar os equipamentos utilizados, e a FUNARBE pela bolsa FUNARBIC/FUNARBE. Ao Prof. Dr. Dorgival Olavo Guedes Neto pelas considerações sobre o trabalho.

7 Contribuição dos Autores

- Gustavo de P. Avelar: Pesquisas bibliográficas, projeto e execução de experimentos, aquisição de dados, configuração do ambiente distribuído, implementação do método de seleção de atributos, análise e interpretação de dados e resultados, elaboração do manuscrito.

- Murilo C. Naldi: Concepção e design do estudo, projeto de experimentos, análise e interpretação dos dados e resultados, elaboração do manuscrito, revisão crítica, aquisição de financiamento.

Referências

1 RUSSOM, P. et al. Big data analytics. *TDWI Best Practices Report, Fourth Quarter*, Renton, WA, USA, p. 1–35, 2011.

2 MANYIKA, J. et al. Big data: The next frontier for innovation, competition, and productivity. McKinsey Global Institute San Francisco, San Francisco, CA, USA, 2011.

3 DEAN, J.; GHEMAWAT, S. Mapreduce: Simplified data processing on large clusters. *Communications of the ACM*, ACM, New York, NY, USA, v. 51, n. 1, p. 107–113, jan. 2008. ISSN 0001-0782. Disponível em: <http://doi.acm.org/10.1145/1327452.1327492>. Acesso em: 2 jun. 2017.

4 WHITE, T. *Hadoop: The definitive guide*. Sebastopol, CA, USA: O'Reilly Media, Inc., 2012.

5 ZAHARIA, M. et al. Spark: cluster computing with working sets. In: 2ND USENIX CONFERENCE ON HOT TOPICS IN CLOUD COMPUTING, 2., 2010, Boston, MA, USA. *Anais...* Berkeley, CA, USA: USENIX Association, 2010. v. 10, p. 10.

6 TAN, P.-N.; STEINBACH, M.; KUMAR, V. *Introdução ao datamining: mineração de dados*. Rio de Janeiro, Brazil: Ciência Moderna, 2009.

7 JAIN, A. K.; DUBES, R. C. et al. *Algorithms for clustering data*. Upper Saddle River, NJ, USA: Prentice hall Englewood Cliffs, 1988. v. 6.

8 JOLLIFFE, I. Principal component analysis. In: ____. *Wiley StatsRef: Statistics Reference Online*. John Wiley & Sons, Ltd, 2014. ISBN 9781118445112. Disponível em: <http://dx.doi.org/10.1002/9781118445112.stat06472>. Acesso em: 20 mai. 2017.

9 SOLÉ-CASALS, J. et al. Improving a leaves automatic recognition process using pca. In: 2ND INTERNATIONAL WORKSHOP ON PRACTICAL APPLICATIONS OF COMPUTATIONAL BIOLOGY AND BIOINFORMATICS (IWPACBB 2008), 2., 2008,

Salamanca, Spain. *Anais...* New York, NY, USA: Springer Science & Business Media, 2008. v. 49, p. 243.

10 BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. *The Journal of Machine Learning Research*, JMLR.org, Massachusetts, v. 3, p. 993–1022, 2003.

11 CAI, D.; HE, X.; HAN, J. Document clustering using locality preserving indexing. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, Piscataway, NJ, USA, v. 17, n. 12, p. 1624–1637, 2005.

12 DING, C.; HE, X. K-means clustering via principal component analysis. In: TWENTY-FIRST INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 21., 2004, Banff, Canada. *Anais...* New York, NY, USA: ACM, 2004. (ICML '04), p. 29–. ISBN 1-58113-838-5. Disponível em: <http://doi.acm.org/10.1145/1015330.1015408>. Acesso em: 2 jun. 2017.

13 DING, C.; LI, T. Adaptive dimension reduction using discriminant analysis and k-means clustering. In: 24TH INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 24., 2007, Corvalis, USA. *Anais...* New York, NY, USA: ACM, 2007. p. 521–528.

14 RAMKUMAR, A. S.; POORNA, B. Text document clustering using dimension reduction technique. *International Journal of Applied Engineering Research*, Delhi, India, v. 11, n. 7, p. 4770–4774, 2016.

15 KIM, Y.-I.; JI, Y.-K.; PARK, S. Big text data clustering using class labels and semantic feature based on hadoop of cloud computing. *International Journal of Software Engineering & Its Applications*, South Korea, v. 8, n. 4, 2014.

16 BHARTI, K. K.; SINGH, P. K. Hybrid dimension reduction by integrating feature selection with feature extraction method for text clustering. *Expert Systems with Applications*, Elsevier, Amsterdam, v. 42, n. 6, p. 3105–3114, 2015.

17 PRABHU, P.; ANBAZHAGAN, N. Improving the performance of k-means clustering for high dimensional data set. *International journal on computer science and engineering*, Citeseer, Chennai, India, v. 3, n. 6, p. 2317–2322, 2011.

18 FRAWLEY, W. J.; PIATETSKY-SHAPIRO, G.; MATHEUS, C. J. Knowledge discovery in databases: An overview. *AI magazine*, Menlo Park, USA, v. 13, n. 3, p. 57, 1992.

19 MANNING, C. D.; SCHÜTZE, H. *Foundations of statistical natural language processing*. Cambridge, MA, USA: MIT Press, 1999. v. 999.

20 WALLACH, H. M. Topic modeling: beyond bag-of-words. In: 23RD INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 23., 2006, Pittsburgh, PA, USA. *Anais...* New York, NY, USA: ACM, 2006. p. 977–984.

21 LUHN, H. P. A statistical approach to mechanized encoding and searching of literary information. *IBM Journal of research and development*, IBM, Riverton, NJ, USA, v. 1, n. 4, p. 309–317, 1957.

22 SALTON, G.; BUCKLEY, C. Term-weighting approaches in automatic text retrieval. *Information processing & management*, Elsevier, Amsterdam, Netherlands, v. 24, n. 5, p. 513–523, 1988.

23 SALTON, G.; FOX, E. A.; WU, H. Extended boolean information retrieval. *Communications of the ACM*, ACM, New York, NY, USA, v. 26, n. 11, p. 1022–1036, 1983.

24 OWEN, S. et al. *Mahout in action*. Greenwich, CT, USA: Manning Publications Co., 2011.

25 KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. *Artificial intelligence*, Elsevier, Essex, UK, v. 97, n. 1, p. 273–324, 1997.

26 LIU, H.; YU, L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, Piscataway, NJ, USA, v. 17, n. 4, p. 491–502, 2005.

27 LIU, T. et al. An evaluation on feature selection for text clustering. In: TWENTIETH INTERNATIONAL CONFERENCE ON INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 20., 2003, Washington, DC, USA. *Anais...* Palo Alto, CA, USA: AAAI Press, 2003. v. 3, p. 488–495.

28 EL-KHAIR, I. A. Effects of stop words elimination for arabic information retrieval: a comparative study. *International Journal of Computing & Information Sciences*, Mississauga, Canada, v. 4, n. 3, p. 119–133, 2006.

29 BLANCHARD, A. Understanding and customizing stopword lists for enhanced patent mapping. *World Patent Information*, Elsevier, Amsterdam, v. 29, n. 4, p. 308–316, 2007.

30 FODOR, I. K. *A survey of dimension reduction techniques*. [S.l.]: Technical Report UCRL-ID-148494, Lawrence Livermore National Laboratory, 2002.

31 VENNER, J. *Pro Hadoop*. New York, NY, USA: Apress, 2009.

32 DIXIT, S.; GWAL, N. An implementation of data pre-processing for small dataset. *International Journal of Computer Applications*, Foundation of Computer Science, New York, USA, v. 103, n. 6, 2014.

33 WEISS, S. M.; INDURKHYA, N. *Predictive data mining: a practical guide*. Burlington, MA, USA: Morgan Kaufmann, 1998.

34 KANTARDZIC, M. *Data mining: concepts, models, methods, and algorithms*. New York, NY, USA: John Wiley & Sons, 2011.

35 ZAKI, M. J.; JR, W. M.; MEIRA, W. *Data mining and analysis: fundamental concepts and algorithms*. New York, NY, USA: Cambridge University Press, 2014.

36 SANCTIS, M. D.; BISIO, I.; ARANITI, G. Data mining algorithms for communication networks control: concepts, survey and guidelines. *IEEE Network*, IEEE, New York, NY, USA, v. 30, n. 1, p. 24–29, 2016.

37 SALTON, G.; WONG, A.; YANG, C.-S. A vector space model for automatic indexing. *Communications of the ACM*, ACM, New York, NY, USA, v. 18, n. 11, p. 613–620, 1975.

38 PORTER, M. F. An algorithm for suffix stripping. In: JONES, K. S.; WILLETT, P. (Ed.). *Readings in Information Retrieval*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1997. p. 313–316. ISBN 1-55860-454-5. Disponível em: <http://dl.acm.org/citation.cfm?id=275537.275705>. Acesso em: 15 mai. 2017.

39 MACQUEEN, J. et al. Some methods for classification and analysis of multivariate observations. In: FIFTH BERKELEY SYMPOSIUM ON MATHEMATICAL STATISTICS AND PROBABILITY, 5., 1967, Berkeley, CA, USA. *Anais...* Berkeley, CA, USA: University of California Press, 1967. v. 1, n. 14, p. 281–297.

40 WU, X. et al. Top 10 algorithms in data mining. *Knowledge and Information Systems*, Springer, New York, NY, USA, v. 14, n. 1, p. 1–37, 2008.

41 ESTEVES, R. M.; PAIS, R.; RONG, C. K-means clustering in the cloud–a mahout test. In: IEEE WORKSHOPS OF INTERNATIONAL CONFERENCE ON ADVANCED INFORMATION NETWORKING AND APPLICATIONS (WAINA), 25., 2011, Biopolis, Singapore. *Anais...* Washington DC, USA: IEEE, 2011. p. 514–519.

42 JAIN, A. K.; MURTY, M. N.; FLYNN, P. J. Data clustering: a review. *ACM computing surveys (CSUR)*, ACM, New York, NY, USA, v. 31, n. 3, p. 264–323, 1999.

43 HALL, D.; JURAFSKY, D.; MANNING, C. D. Studying the history of ideas using topic models. In: CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING, 8., 2008, Honolulu, Hawaii. Stroudsburg, USA: Association for Computational Linguistics, 2008. p. 363–371.

44 GRIFFITHS, T. L.; STEYVERS, M. Finding scientific topics. *Proceedings of the National Academy of Sciences*, National Acad Sciences, Washington DC, USA, v. 101, n. suppl 1, p. 5228–5235, 2004.

45 STEYVERS, M.; GRIFFITHS, T. Probabilistic topic models. *Handbook of latent semantic analysis*, Mahwah, NJ, USA, v. 427, n. 7, p. 424–440, 2007.

46 ROBERTS, R. J. PubMed Central: The GenBank of the published literature. *Proceedings of the National Academy of Sciences of the United States of America*, Washington, DC, USA, v. 98, n. 2, p. 381–382, 2001.

47 WEI, C.-H.; KAO, H.-Y.; LU, Z. Pubtator: a web-based text mining tool for assisting biocuration. *Nucleic Acids Research*, Oxford, England, v. 41, 07 2013. Disponível em: <http://nar.oxfordjournals.org/content/early/2013/05/22/nar.gkt441.abstract?keytype=ref&ijkey=mj3EevK5My5SA5D>. Acesso em: 15 mai. 2017.

48 WAGNER, S.; WAGNER, D. *Comparing clusterings: an overview*. Karlsruhe, Germany: Universität Karlsruhe, Fakultät für Informatik Karlsruhe, 2007.

49 MELNYKOV, V.; CHEN, W.-C.; MAITRA, R. Mixsim: An r package for simulating data to study performance of clustering algorithms. *Journal of Statistical Software*, Los Angeles, CA, USA, v. 51, n. 12, p. 1–25, 2012.

50 GREENLAND, S. et al. Statistical tests, p values, confidence intervals, and power: a guide to misinterpretations. *European Journal of Epidemiology*, Dordrecht, Netherlands, v. 31, n. 4, p. 337–350, 2016. ISSN 1573-7284. Disponível em: <http://dx.doi.org/10.1007/s10654-016-0149-3>. Acesso em: 29 mai. 2017.

51 PREL, J.-B. d. et al. Confidence Interval or P-Value? Part 4 of a Series on Evaluation of Scientific Publications. *Dtsch Arztebl International*, Cologne, Germany, v. 106, n. 19, p. 335–339, 2009. Disponível em: <http://www.aerzteblatt.de/int/article.asp?id=64639>. Acesso em: 20 mai. 2017.